REMARKS

This is in response to the Office Action dated March 27, 2006. Reconsideration of this application is respectfully requested in view of this response.

STATUS OF CLAIMS

Claims 1-33 are pending.

Claims 1-33 are rejected under 35 U.S.C. § 103(a) as being unpatentable over non-patent literature "Least Expected Cost Query Optimization, An Exercise in Utility" (Chu) in view of U.S. 5,301,317 (Lohman).

OVERVIEW OF CLAIMED INVENTION

The present invention provides for a compilation time estimator (COTE) and a method implemented in the COTE for estimating optimization complexity by estimating the number of distinct join plans that will be generated, rather than the number of join sequences. The join enumerator in an optimizer is reused to iterate all the join pairs, but plan generation is bypassed. A small number of differentiating properties are accumulated during enumeration to calculate the number of generated plans for each enumerated join. Hence, instead of estimating the number of join sequences, the present invention's COTE estimates the number of join plans (a finer granularity). Since the cost of generating a join plan is much more uniform than that of a join sequence, the estimator provides more accurate compilation time estimation.

The method of the present invention comprises the steps of: (a) reusing the existing join enumerator in an optimizer to iterate through all the possible join sequences, but bypassing the

expensive plan generation step (to avoid the overhead of space allocation and cost estimation, etc); (b) determining a small number of differentiating properties (e.g., properties used to distinguish plans) that affect the number of plans generated for each join sequence and using those factors to calculate the number of generated plans; and (c) estimating the compilation time from the number of generated plans using a regression model (e.g., a linear regression model on the number of generated plans for each type of join method).

The compilation time estimator (COTE) of the present invention can be used for many other applications. For example, the COTE is useful in evaluating the need for mid-query reoptimization, in which an optimizer tries to generate a new plan in the middle of execution if a significant cardinality discrepancy is discovered. Since reoptimization itself takes time, the decision on whether to reoptimize or not is better made by comparing the execution cost of the remaining work with the estimated time to recompile.

Estimating the compilation time is also very useful for workload analysis tools. Examples of these tools are advisors for indexes, materialized views, and partitioning that have been built on top of commercial database systems. All these tools spend most of their time compiling (but not executing) a large number of queries in the input workload as part of their tuning analysis, and run for hours or even days, depending on the workload. A compilation time estimator (COTE) is used to forecast how long such a tool would take to finish and possibly to show the progress of the tool as well.

REJECTIONS UNDER 35 U.S.C. § 103

Claims 1-33 are rejected under 35 U.S.C. § 103(a) as being unpatentable over non-patent literature "Least Expected Cost Query Optimization, An Exercise in Utility" (Chu) in view of U.S. 5,301,317 (Lohman). To be properly rejected under 35 U.S.C. § 103(a), the cited references have to provide for each and every element of the rejected claims. Applicants respectfully submit that the combination of the Chu et al. paper in view of the Lohman et al. reference fails to teach many of the features of claims 1-33.

Chu et al. provide for query optimization algorithms directed towards finding the plan of least expected cost (LEC). According to Chu et al., their plan chooses the plan of the least expected cost instead of the plan of least cost given expected values of the parameters.

The US patent to Lohman et al. (5,301,317) discloses a system providing automatic adjustment of resources devoted to query optimization according to estimated query execution time. The described system permits the query optimizer to automatically trade off the time spent estimating the execution cost of alternate query execution plans against the potential savings in execution time that one of those alternate plans may yield. The number of alternate plans considered is adjusted by selecting compile-time parameters and heuristic criteria for limiting the primitive database operators used in the alternate plans, thereby establishing a new search space. The parameters and criteria are adjusted according to the estimate of execution cost for the optimal plan from a first search space.

Applicants agree with the Examiner's statement on page 3 of the office action of 03/27/2006 that the Chu et al. reference fails to teach the iterative steps of independent claim 1. However, Applicants respectfully disagree with the Examiner's statement that the Lohman et al. reference (it should be noted that Guy M Lohman is also an inventor in the pending application) remedies this shortcoming.

Specifically, the Examiner cites figure 8 and accompanying description in column 13, lines 30-35 of Lohman et al. as teaching the iterative feature of claim 1. Figures 7 and 8 of Lohman et al. provide flow charts of the preferred optimization effort adjustment procedure, wherein the process proceeds from one predetermined space to the next, halting the process as soon as the expected improvement and execution time is found to fall short of the requisite threshold of improvement over the estimated execution time of the best plan found so far. Specifically, Applicants wish to note that the only mention of a "join" in figure 8 are the steps to select and revise "internal feasible join filter criteria according to search space sizing parameters". Applicants respectfully submit that this step, shown in both figures 7 and 8 of Lohman et al., merely describes the selection and revision of the "internal feasible join filter criteria" and neither explicitly nor implicitly mentions iterating through possible join pairs for a query or, for each join pair, identifying a set of differentiating properties and using the identified set differentiating properties to calculate number of join plans, wherein the compilation time is estimated from the calculated number of join plans for each type of join method.

Further, column 13, lines 30-35 merely restates the above with the addition that the search spaces are enlarged by replacing the feasible join criteria, a teaching that neither

anticipates nor renders obvious claim 1's feature of iterating through possible join pairs for a query or, for each join pair, identifying a set of differentiating properties and using the identified set differentiating properties to calculate number of join plans, wherein the compilation time is estimated from the calculated number of join plans for each type of join method.

Applicants are unsure how the Examiner is interpreting Lohman et al.'s feature of selecting and revising internal feasible join criteria to read on applicants' feature of iterating through feasible join pairs for a query or for each join pair, identifying a set of differentiating properties and using the identified set differentiating properties to calculate number of join plans.

Applicants agree with the Examiner's statement on page 6 of the office action of 03/27/2006 that the Chu et al. reference fails to teach claim 11's feature of reusing a join enumerator to estimate compilation time of the query optimizer. However, Applicants respectfully disagree with the Examiner's statement that the Lohman et al. reference remedies this shortcoming.

Specifically, the Examiner cites figure 8, column 11, lines 36-43, and column 13, lines 30-35 as teaching the feature of reusing a join enumerator to estimate compilation time of the query optimizer, wherein the join enumerator iterates through possible join pairs for a query. As mentioned above, figure 7 and 8 of Lohman et al. merely provide flow charts of the preferred optimization effort adjustment procedure, wherein the process proceeds from one predetermined space to the next, halting the process as soon as the expected improvement and execution time is found to fall short of the requisite threshold of improvement over the estimated execution time of

the best plan found so far. Also, as mentioned above, column 13, lines 30-35 merely mentions the flow charts of figure 7 and 8, where the only recitation of a join operation is the selection and revision the "internal join filter criteria" for a given search space. Applicants respectfully submit that this feature cannot be equated to the claim 11's feature of "reusing a join enumerator to estimate compilation time of a query optimizer, said join enumerator iterating through possible join pairs for a query."

Further, the Examiner relies on the Lohman et al. reference as teaching the feature of estimating compilation time via a regression model as follows:

$$T = T_{inst} \times \sum (C_t \times P_t)$$

wherein T is a machine-dependent parameter representing time per instruction, C_t is a constant representing number of instructions to generate a join plan of type t, and P_t is an estimated number of join plans of type t. For support, the Examiner cites column 5, lines 47-53 of Lohman et al. as teaching the above-presented equation pf claim 11. Applicants respectfully disagree with the Examiner's statement, as the citation neither explicitly nor implicitly provides for the equation of claim 11.

Column 5, lines 47-53 of the Lohman et al. reference merely teaches that the product of the "estimated number [of] plans" and the "single plan evaluation time" yields a rough estimate of the evaluation cost for the entire larger search space. Applicants respectfully remind the Examiner that the "estimated number [of] plans", as described in Lohman et al., is NOT the same as claim 11's "estimated number of join plans of type t".

Further, the Examiner equates the "threshold" mentioned in column 5, lines 50-53 to C_t of claim 11. The "threshold" mentioned in column 5, lines 50-53 of Lohman et al. represents the condition that the "larger search space is then evaluated (searched) only if the initial estimated execution cost exceeds the estimated space evaluation time by some threshold". Applicants respectfully assert that the threshold of Lohman et al. (over which the execution cost exceeds the estimated space evaluation time) cannot be explicitly or implicitly equated to claim 11's C_t representing the number of instructions to generate a join plan of type t.

Additionally, the Examiner equates claim 11's machine-dependent parameter representing time per instruction to Lohman's single plan evaluation time. Applicants respectfully submit that Lohman's single plan evaluation time merely corresponds to the time to evaluate a single plan whereas Applicants' machine-dependent parameter represents time per instruction. Hence, Applicants respectfully assert that Lohman et al.'s "single plan evaluation time" cannot be explicitly or implicitly equated to Applicants' machine-dependent parameter representing time per instruction.

Also, Applicants respectfully assert that the Examiner has failed to show where in the Lohman et al. reference or the Chu et al. reference the specific calculation, via a regression model, of the compilation time as per $T = T_{inst} \times \sum (C_t \times P_t)$ is shown.

If the Examiner still feels that such a specific recitation for the above-mentioned computation (via a regression model) of claim 11 is found in either Lohman et al. or Chu et al., Applicants' wish to emphasize that it is the duty of the examiner to specifically point out

limitations with respect to each and every claim element such that applicants' are aware of how the examiner is applying a reference in a rejection. Specifically, §1.104(c)(2) of Title 37 of the Code of Federal Regulations and section 707 of the M.P.E.P explicitly states that "the particular part relied on must be designated" and "the pertinence of each reference, if not apparent, must be clearly explained and each rejected claim specified".

The above-mentioned arguments with respect to independent claims 1 and 11 substantially apply to independent claims 16, 22, and 23 as they recite many similar features. Further, the above-mentioned arguments with respect to the independent claims substantially apply to the dependent claims as they inherit all the limitations of the claim from which they depend.

Further, with respect to independent claim 28, the Examiner states that he interprets Lohman et al.'s mention "reducing the number of feasible plans" as bypassing plan generation. Applicants respectfully disagree with this statement. Specifically, Applicants respectfully assert that column 6, lines 60-63 of Lohman et al. merely state that in the instance that the queries are complex the feasibility criteria may be made restrictive to reduce the number of feasible plans. Applicants are unsure how the Examiner is interpreting the reduction in the number of feasible plans to read on the feature of "bypassing plan generation and reusing a join enumerator" of the query optimizer to identify number of joins. In fact, Applicants assert that the preferred embodiments described in figure 7 and 8 of Lohman et al. specifically describe a step to "generate plans for search space S(1)", which is in stark contrast to Applicants claim 28.

If the Examiner still feels that such a recitation bypassing plan generation is found in either Lohman et al. or Chu et al., Applicants' wish to emphasize that it is the duty of the examiner to specifically point out limitations with respect to each and every claim element such that applicants' are aware of how the examiner is applying a reference in a rejection. Specifically, §1.104(c)(2) of Title 37 of the Code of Federal Regulations and section 707 of the M.P.E.P explicitly states that "the particular part relied on must be designated" and "the pertinence of each reference, if not apparent, must be clearly explained and each rejected claim specified".

Based on the arguments presented above, Applicants respectfully request the Examiner to remove the rejections with respect to the pending claims and request the allowance thereof.

SUMMARY

As has been detailed above, none of the references, cited or applied, provide for the specific claimed details of applicants' presently claimed invention, nor renders them obvious. It is believed that this case is in condition for allowance and reconsideration thereof and early issuance is respectfully requested.

As this response has been timely filed within the set time period, no request for extension of time or associated fee is required. However, the Commissioner is hereby authorized to charge any deficiencies in the fees provided to Deposit Account No. 09-0441.

If it is felt that an interview would expedite prosecution of this application, please do not hesitate to contact applicants' representative at the below number.

Respectfully submitted,

Ramraj Soundararajan
Registration No. 53,832

1725 Duke Street
Suite 650
Alexandria, Virginia 22314
(703) 838-7683
May 12, 2006